

A Framework to Adjust Dependency Measure Estimates for Chance

Simone Romano* Nguyen Xuan Vinh* James Bailey* Karin Verspoor*

Abstract

Estimating the strength of dependency between two variables is fundamental for exploratory analysis and many other applications in data mining. For example: non-linear dependencies between two continuous variables can be explored with the Maximal Information Coefficient (MIC); and categorical variables that are dependent to the target class are selected using Gini gain in random forests. Nonetheless, because dependency measures are estimated on finite samples, the interpretability of their quantification and the accuracy when ranking dependencies become challenging. Dependency estimates are not equal to 0 when variables are independent, cannot be compared if computed on different sample size, and they are inflated by chance on variables with more categories. In this paper, we propose a framework to adjust dependency measure estimates on finite samples. Our adjustments, which are simple and applicable to any dependency measure, are helpful in improving interpretability when quantifying dependency and in improving accuracy on the task of ranking dependencies. In particular, we demonstrate that our approach enhances the interpretability of MIC when used as a proxy for the amount of noise between variables, and to gain accuracy when ranking variables during the splitting procedure in random forests.

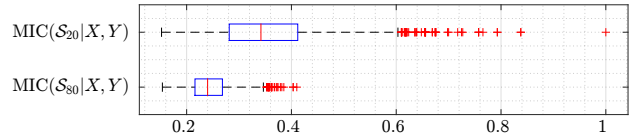
1 Introduction

Dependency measures $\mathcal{D}(X, Y)$ are employed in data mining to assess the strength of the dependency between two continuous or categorical variables X and Y . If the variables are continuous, we can use Pearson's correlation to detect linear dependencies, or use more sophisticated measures, such as the Maximal Information Coefficient (MIC) [1] to detect *non*-linear dependencies. If the variables are categorical we can use the well known mutual information (a.k.a. information gain) or the Gini gain [2]. Dependency measures are ubiquitously used: to infer biological networks [1], for variable selection for classification and regression tasks [3], for clustering comparisons and validation [4], as splitting

criteria in random forest [5], and to evaluate classification accuracy [6], to list a few.

Nonetheless, there exist a number of problems when the dependency $\mathcal{D}(X, Y)$ is estimated with $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ on a data sample \mathcal{S}_n of n data points: *a)* even if the population value $\mathcal{D}(X, Y) = 0$ when X and Y are statistically independent, estimates have a high chance to be bigger than 0 when n is finite; *b)* when comparing pairs of variables which share the same fixed population value $\mathcal{D}(X, Y)$, estimates are still dependent on the sample size n and the number of categories of X and Y . These issues diminish the utility of dependency measures on *quantification* tasks. For example, MIC was proposed in [1] as a proxy of the amount of noise on the functional dependence between X and Y : it should “provide a score that roughly equals the coefficient of determination R^2 of the data relative to the regression function”, which is 0 under complete noise and 1 in noiseless scenarios. Nonetheless, MIC is not equal to 0 under complete noise, and MIC values are not comparable if computed on samples of different size n because of the use of different datasets or in the case of variables with missing values:

EXAMPLE 1. *Given two uniform and independent variables X and Y in $[0, 1]$, the population value of MIC is 0 but the estimates $\text{MIC}(\mathcal{S}_{20}|X, Y)$ on 20 data points are higher than $\text{MIC}(\mathcal{S}_{80}|X, Y)$ on 80 data points. On average, they achieve the values of 0.36 and 0.25 respectively. The user expects this value to be 0. The following box plots show estimates for 10,000 simulations.*



The example above shows that the estimated MIC does not have zero baseline for finite samples. The zero baseline property is well known in the clustering community [4], nonetheless this property does not hold for many dependency measures used in data mining.

Problems also arise when *ranking* dependencies on a finite data sample. For example, if Gini gain is used to rank the dependency between variables to the target class in random forests [5], variables with more

*CIS Department, The University of Melbourne, Australia
 {simone.romano, vinh.nguyen, baileyj, karin.verspoor}
 @unimelb.edu.au

categories have more chances to be ranked higher:

EXAMPLE 2. *Given a variable X_1 with two categories and a variable X_2 with one more category which are both independent of the target binary class Y , both the population value of Gini gain between X_1 and Y , and the population value between X_2 and Y are equal to 0. However, when Gini gain is estimated on 100 data points the probability of $\text{Gini}(\mathcal{S}_{100}|X_2, Y)$ being greater than $\text{Gini}(\mathcal{S}_{100}|X_1, Y)$ is equal to 0.7. The user expects 0.5 given that X_1 and X_2 are equally uninformative to Y .*

It is common practice to use the p -value of Gini gain to correct this bias [7]. Nonetheless, we will shortly see that p -values are effective only when the population value of a dependency measure is 0.

In this paper, we identify that the issues discussed in Example 1 and 2 are due to inflated estimates arising from finite samples. Statistical properties of the distribution of the dependency measure estimator $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ under independence of X and Y can be used to adjust these estimates. The challenge is to formalize a general framework to adjust dependency measure estimates which also addresses the shortcomings of the use of p -values. We make the following contributions:

- We identify common biases of dependency measure estimates due to finite samples;
- We propose a framework to adjust estimates $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ which is simple, yet applicable to many dependency measures because it only requires to use the distribution of the estimator when X and Y are independent;
- We experimentally demonstrate that our adjustments improve interpretability when quantifying dependency (e.g., when using MIC as a proxy of the amount of noise) and accuracy when ranking dependencies (e.g., when using Gini gain in random forests).

2 Background

Dependency measures $\mathcal{D}(X, Y)$ are defined on the joint distribution (X, Y) . In data mining applications, they are estimated with $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ on a finite sample $\mathcal{S}_n = \{(x_k, y_k)\}$ of n data points. If variables are continuous, we can compute the amount of linear dependency with the squared Pearson’s correlation coefficient:

$$(2.1) \quad r^2(\mathcal{S}_n|X, Y) \triangleq \frac{\left(\sum_{k=1}^n (x_k - \bar{x})(y_k - \bar{y})\right)^2}{\sum_{k=1}^n (x_k - \bar{x})^2 \sum_{k=1}^n (y_k - \bar{y})^2}$$

with $\bar{x} = \frac{1}{n} \sum_{k=1}^n x_k$ and $\bar{y} = \frac{1}{n} \sum_{k=1}^n y_k$. If we are interested in *non*-linear relationships, we can employ the Maximal Information Coefficient (MIC) [1].

$\text{MIC}(\mathcal{S}_n|X, Y)$ is estimated as the maximum normalized mutual information across all the possible grids superimposed on the sample \mathcal{S}_n to estimate the joint distribution of X and Y . When the variables are categorical, the mutual information or $\text{Gini}(\mathcal{S}_n|X, Y)$ can be directly estimated using the joint empirical probability distribution between X and Y on the sample \mathcal{S}_n . See Appendix A in the supplement for formal definitions.

There are three important applications of dependency measures between two variables [8]:

Detection: Test for the presence of dependency. For example, assess if there exists any dependence between bacterial species that colonize the gut of mammals [1];

Quantification: Summarization of the amount of dependency in an interpretable fashion. For example, when MIC is used as a proxy of the amount of noise in a relationship [1];

Ranking: Sort the relationships of different variables based on the strength of their dependency. For example, when Gini gain is used to rank predictive variables to the target class in random forests [5].

We saw in Examples 1 and 2 that when it comes to estimating dependency on data samples via $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ the interpretability of *quantification* and accuracy of *ranking* become challenging. We claim that both tasks can take advantage of the distribution of $\hat{\mathcal{D}}$ under the following null hypothesis:

DEFINITION 2.1. $\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)$ is the distribution of $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ on a sample \mathcal{S}_n under the **null hypothesis** that X is statistically independent of Y .

This null hypothesis is commonly exploited only in *detection* tasks where the distribution $\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)$ is computed under the null and a p -value is computed to filter out false discoveries [1]. Nonetheless, this null can be used also to aid *quantification* and *ranking*. The challenges are to identify the distribution under the null for a particular dependency measure, and to employ it in a framework to perform adjustments to the estimates. Here we discuss the use of this null hypothesis in previous research.

2.1 Use of the Null for Quantification. To our knowledge the first instance of a systematic approach using the null distribution $\hat{\mathcal{D}}_0$ to achieve interpretability in quantification was proposed in the 1960 with the κ coefficient of inter-annotator agreement [6]. The amount of agreement $A(\mathcal{S}_n|X, Y)$ (dependency) between two annotators X and Y on a sample of n items can be adjusted for chance by subtracting its expected value $E[A_0(\mathcal{S}_n|X, Y)]$ under the null hypothesis of independence between annotators. The κ coefficient is obtained

by normalization via division of its maximum value $\max A = 1$ to obtain an adjusted dependency measure in the range $[0, 1]$:

$$(2.2) \quad \kappa(\mathcal{S}_n|X, Y) = \frac{A(\mathcal{S}_n|X, Y) - E[A_0(\mathcal{S}_n|X, Y)]}{1 - E[A_0(\mathcal{S}_n|X, Y)]}$$

Other notable examples are the Adjusted Rand Index (ARI) and the Adjusted Mutual Information (AMI) [4]. We argue that this approach should be applied to many other dependency measures estimators $\hat{\mathcal{D}}$ because it improves interpretability by guaranteeing a zero baseline to $\hat{\mathcal{D}}$. Moreover, we will shortly see that it helps in comparing estimates on different samples \mathcal{S}_n .

2.2 Use of the Null for Ranking. In the decision tree community, it is very well known that when selecting the most dependent variable X to the target class Y , variables available on a small number of samples n or with many categories tend to be chosen more often. Indeed, it has been shown that an unbiased selection can be obtained if the p -value [7] of a dependency estimate or its standardized version [9] is used rather than its raw value. Nonetheless, these techniques are unbiased only under the null hypothesis and not unbiased in general. Indeed, in the next sections we will see that their use actually yields bias towards variables induced on bigger n or with fewer categories. This behavior has been overlooked in the decision tree community.

3 Adjusting Estimates for Quantification

To guarantee good interpretability in quantification tasks, dependency measure estimates should be equal to 0 on average when X and Y are independent, and their values should be comparable on average across different data samples of different size. More formally we want:

PROPERTY 3.1. (ZERO BASELINE) *If X and Y are independent then $E[\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)] = 0$ for all n .*

PROPERTY 3.2. (QUANTIFICATION UNBIASEDNESS) *If $\mathcal{D}(X_1, Y_1) = \mathcal{D}(X_2, Y_2)$ then $E[\hat{\mathcal{D}}(\mathcal{S}_n|X_1, Y_1)] = E[\hat{\mathcal{D}}(\mathcal{S}_m|X_2, Y_2)]$ for all n and m .*

We saw in Example 1 that MIC does not satisfy either property. Therefore, we propose an adjustment that can be applied to MIC and in general to any dependency estimator $\hat{\mathcal{D}}$:

DEFINITION 3.1. (ADJUSTMENT FOR QUANTIFICATION)

$$A\hat{\mathcal{D}}(\mathcal{S}_n|X, Y) \triangleq \frac{\hat{\mathcal{D}}(\mathcal{S}_n|X, Y) - E[\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)]}{\max \hat{\mathcal{D}}(\mathcal{S}_n|X, Y) - E[\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)]}$$

is the adjustment of $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$, where $\max \hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ and $E[\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)]$ are respec-

tively the maximum of $\hat{\mathcal{D}}$, and its expected value under the null.

$A\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$ has always zero baseline (Property 3.1) being 0 on average when X and Y are independent, and attains 1 as maximum value. This adjustment can be applied to r^2 and MIC to increase their interpretability when they are used as proxies of the amount of noise in a linear relationship and a functional relationship respectively. We just have to identify their distribution on the sample \mathcal{S}_n under the null:

- $r_0^2(\mathcal{S}_n|X, Y)$: follows a Beta distribution with parameters $\frac{1}{2}$ and $\frac{n-2}{2}$ [10];
- $\text{MIC}_0(\mathcal{S}_n|X, Y)$: this distribution can be computed using $s = 1, \dots, S$ Monte Carlo permutations $\text{MIC}_0^{(s)}$ of MIC [1]. See Appendix A.1.

Therefore the adjusted Pearson's correlation squared r^2 and the adjusted MIC are:

$$(3.3) \quad Ar^2(\mathcal{S}_n|X, Y) = \frac{r^2(\mathcal{S}_n|X, Y) - \frac{1}{n-1}}{1 - \frac{1}{n-1}}$$

$$(3.4) \quad \text{AMIC}(\mathcal{S}_n|X, Y) = \frac{\text{MIC}(\mathcal{S}_n|X, Y) - \text{EMIC}_0}{1 - \text{EMIC}_0}$$

where $E[r_0^2(\mathcal{S}_n|X, Y)] = \frac{1}{n-1}$ and $\text{EMIC}_0 = \frac{1}{S} \sum_{s=1}^S \text{MIC}_0^{(s)}$. EMIC_0 converges to $E[\text{MIC}_0(\mathcal{S}_n|X, Y)]$ at the limit of infinite permutations. However, good estimation accuracy can be obtained even with few permutations because of the law of large numbers [11].

In the next section we will see how our adjustments satisfy Property 3.1 and Property 3.2.

3.1 Experiments with Pearson Correlation and MIC. We aim to experimentally verify the zero baseline Property 3.1 and that our adjustment in Definition 3.1 enables better *interpretability*. We generate a linear relationship between a uniformly distributed X in $[0, 1]$ and Y on $n = 30$ points adding different percentages of white noise. We compare r^2 and Ar^2 . Each white noise level is obtained by substituting a given percentage of points from the relationship and assigning to the Y coordinate a random value in $[0, 1]$. Figure 1 shows the average r^2 and Ar^2 for 2,000 simulated relationships with a given percentage of white noise: r^2 is not zero on average when the amount of noise is 100% (last plot on the right). On the other hand, Ar^2 is very close to zero when there is complete noise and it fully exploits its range of values from one to zero, mapping the domain from 0% to 100% noise. This yields more interpretability and enables Ar^2 to be used as a *proxy to quantify the amount of noise in linear relationships*.

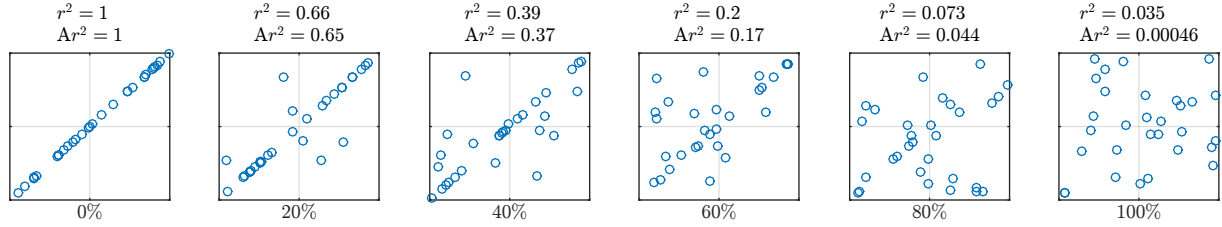


Figure 1: Average value of r^2 and Ar^2 for different percentages of white noise. *Linear* relationship between X and Y induced on $n = 30$ points in $[0, 1] \times [0, 1]$. Ar^2 becomes zero on average on 100% noise enabling a more interpretable range of variation.

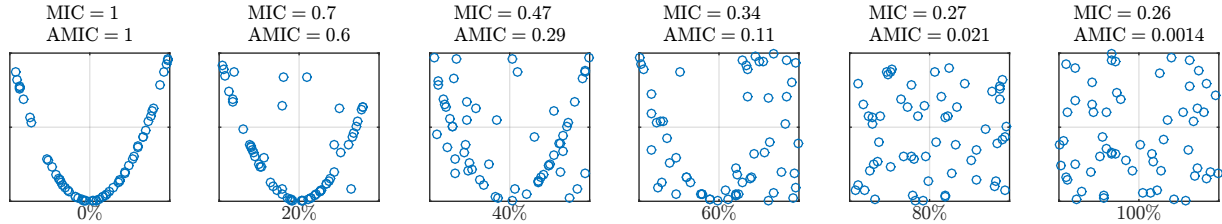


Figure 2: Average value of MIC and AMIC for different percentages of white noise. *Quadratic* relationship between X and Y induced on $n = 60$ points in $[0, 1] \times [0, 1]$. AMIC becomes zero on average on 100% noise enabling a more interpretable range of variation.

Similarly, we generated a quadratic relationship between X and Y in $[0, 1] \times [0, 1]$ on $n = 60$ points with different levels of noise to compare MIC and AMIC. Figure 2 shows that the value of MIC computed with default parameters [1], is about 0.26 on average for complete noise. AMIC computed with $S = 30$ permutations, is instead very close to zero and it exploits better its range of values from one to zero. AMIC is more interpretable than MIC and might be used more intuitively as a *proxy for the amount of noise in a functional relationship*.

The average value of a dependency estimator should not be biased with regards to the sample \mathcal{S}_n as stated in Property 3.2. In Figure 3, we show that r^2 and MIC suffer from this problem: their estimates are higher on average when n is smaller. Figure 3 shows the average value of raw and adjusted measures on 2,000 simulations for different levels of noise and sample size n : r^2 and Ar^2 are compared on linear relationships; MIC and AMIC are compared on linear, quadratic, cubic, and 4th root relationships. Neither the zero baseline Property 3.1 nor the quantification unbiasedness Property 3.2 is verified for the raw measures r^2 and MIC, shown respectively in Figure 3(a) and 3(c). Instead, Ar^2 and AMIC in Figure 3(b) and 3(d), satisfy both properties: they have zero baseline and their average value is not biased with regards to the sample size n . We claim that these properties improve *interpretability* when quantifying dependency and *enhance equitability* for MIC [1].

4 Adjusting Estimates for Ranking

When the task is ranking dependencies according to their strength, dependencies induced on smaller sample size n or on variables with more categories have more chances to be ranked higher as shown in Example 2 for Gini gain. This issue is due to inflated estimates due to finite samples. Indeed, r^2 and MIC suffer from the same problem.

Consider this experiment: we generate five samples \mathcal{S}_n with $n = [20, 40, 60, 80, 100]$ to simulate different amount of missing values for a joint distribution (X, Y) where X and Y are independent. For each sample, we compute $r^2(\mathcal{S}_n|X, Y)$, we select \mathcal{S}_n that achieves the highest value, and iterate this process 10,000 times. Given that the population value $\rho^2(X, Y) = 0$ for all samples, all samples should have equal chances to maximize the r^2 . However, Figure 4 shows that \mathcal{S}_{20} has higher chances to maximize r^2 . *This implies that dependencies estimated on samples with missing values have higher chances to be ranked higher in terms of strength.*

We would like that dependencies which share the same population value for \mathcal{D} had the same chances to maximize the dependency estimate $\hat{\mathcal{D}}$ even if estimated on different samples. More formally:

PROPERTY 4.1. (RANKING UNBIASEDNESS) *If $\mathcal{D}(X_1, Y_1) = \mathcal{D}(X_2, Y_2) = \dots = \mathcal{D}(X_K, Y_K)$ then the probability of $\hat{\mathcal{D}}(\mathcal{S}_{n_i}|X_i, Y_i)$ being equal or greater than any $\hat{\mathcal{D}}(\mathcal{S}_{n_j}|X_j, Y_j)$ is $\frac{1}{K}$ for all $n_i, n_j, 1 \leq i \neq j \leq K$.*

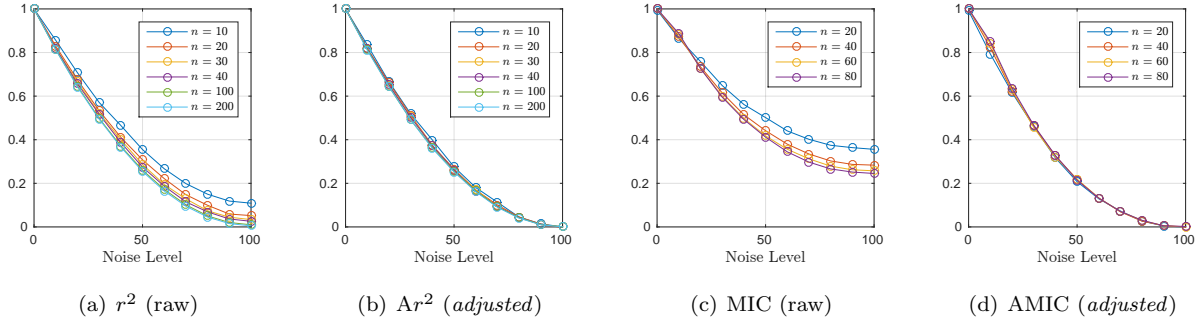


Figure 3: Average value of r^2 , Ar^2 , MIC, and AMIC on different amount of noise and different sample size n . Raw measures show higher values for smaller n on average. Instead, Property 3.2 of unbiasedness with regards to n is empirically verified for *adjusted* measures.

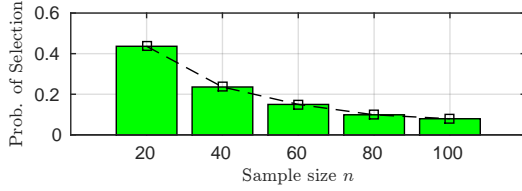


Figure 4: Probability to select the sample \mathcal{S}_n with $n = [20, 40, 60, 80, 100]$ according $r^2(\mathcal{S}_n|X, Y)$ fixing the population value $\rho^2(X, Y) = 0$. The relationship with $n = 20$ has more chances to be ranked higher.

For example in Figure 4 we would like constant probability of selection equal to $\frac{1}{5} = 0.20$. Property 4.1 is useful to achieve *higher accuracy* when the task is ranking the pair of variables that show the stronger relationship.

Biases in ranking are well known in the decision tree community [7] as shown in Example 2. Distributional properties of the raw dependency measure have to be employed to adjust for biases in ranking. For example, ranking according to p -values or standardized measures are possible solutions [7, 9]. They both quantify if the estimate $\hat{\mathcal{D}}$ is statistically significant. Here we extend the standardization technique to any dependency measure estimate $\hat{\mathcal{D}}$ to employ it for unbiased ranking:

DEFINITION 4.1. (STANDARDIZATION FOR RANKING)

$$S\hat{\mathcal{D}}(\mathcal{S}_n|X, Y) \triangleq \frac{\hat{\mathcal{D}}(\mathcal{S}_n|X, Y) - E[\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)]}{\sqrt{\text{Var}(\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y))}}$$

is the standardized $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$, where $E[\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)]$ and $\text{Var}(\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y))$ are, respectively, the expected value and the variance of $\hat{\mathcal{D}}$ under the null.

Nonetheless, it is very difficult to satisfy the ranking unbiasedness Property 4.1 just with $S\hat{\mathcal{D}}$. Therefore we also define an adjustment to dependency measures

whose bias can be tuned according to a parameter α . This is particularly useful when α can be tuned with cross-validation, e.g. in random forests.

DEFINITION 4.2. (ADJUSTMENT FOR RANKING)

$$A\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)(\alpha) \triangleq \hat{\mathcal{D}}(\mathcal{S}_n|X, Y) - q_0(1 - \alpha)$$

is the adjustment at level $\alpha \in (0, 1]$ of $\hat{\mathcal{D}}(\mathcal{S}_n|X, Y)$, where $q_0(1 - \alpha)$ is the $(1 - \alpha)$ -quantile of $\hat{\mathcal{D}}_0(\mathcal{S}_n|X, Y)$ under the null: i.e., $P(\hat{\mathcal{D}}(\mathcal{S}_n|X, Y) \leq q_0(1 - \alpha)) = 1 - \alpha$.

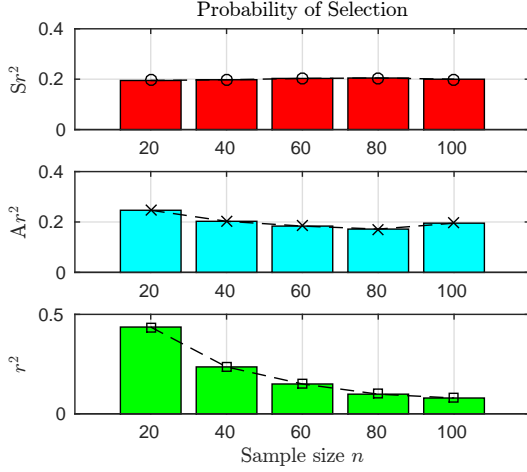
At a fixed significance level α , the quantile $q_0(1 - \alpha)$ induces more penalization when the estimate is not statistically significant. With regards to Example 2, fixing $\alpha = 0.05$ we penalize the variable X_1 and the variable X_2 by $q_0(0.95)$ equal to 0.036 and 0.053 respectively. The latter variable gets penalized more because it is less statistically significant having more categories. In contrast, $S\hat{\mathcal{D}}$ fixes the amount of penalization based on statistical significance and does not allow to tune the bias during ranking. In the next section we aim to show the shortcomings of raw measures and standardized measures for ranking tasks.

4.1 Ranking Biases of Raw and Standardized Measures.

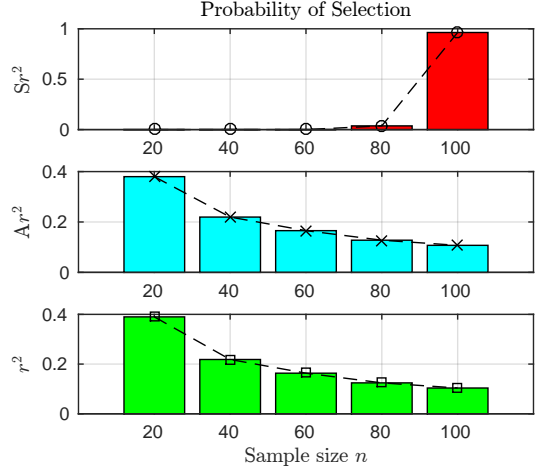
We use r^2 and its adjusted versions in a case study: Ar^2 is defined as per Eq. (3.3), $Ar^2(\alpha) = r^2 - q_0(1 - \alpha)$ where $q_0(1 - \alpha)$ is computed with the Beta distribution (see Section 3), and the standardized r^2 is defined as:

$$(4.5) \quad Sr^2(\mathcal{S}_n|X, Y) = \frac{r^2(\mathcal{S}_n|X, Y) - \frac{1}{n-1}}{\sqrt{\frac{2(n-2)}{(n-1)^2(n+1)}}}$$

We do not evaluate p -values because their use is equivalent to the use of standardized measures which are also much easier to compute.



(a) X independent of Y ($\rho^2 = 0$).



(b) X linearly related to Y with 10% white noise ($\rho^2 > 0$).

Figure 5: Probability to select the sample \mathcal{S}_n induced on $n = [20, 40, 60, 80, 100]$ according adjusted measures: Sr^2 satisfies the ranking unbiasedness Property 4.1 when $\rho^2 = 0$ but not when $\rho^2 > 0$. All measures show to be biased in the latter case: it is difficult to satisfy Property 4.1 in general.

We perform similar experiments as in the previous section: we fix the population value for a dependency and compute estimates on different samples \mathcal{S}_n to compute their probability of selection. We select samples according r^2 , Ar^2 , and Sr^2 . Figure 5(a) shows the probability of selection of different samples at fixed population value $\rho^2 = 0$. We can clearly see that the ranking unbiasedness Property 4.1 is satisfied if we use Sr^2 (top plot). On the other hand the sole adjustment for quantification Ar^2 is not enough to remove r^2 bias towards small n . Nonetheless, Figure 5(b) shows that if we generate a linear relationship between X and Y with 10% white noise (i.e., ρ^2 is fixed to a value greater than 0), Sr^2 is biased towards big n . This is because we prefer statistically significant relationships. This phenomena might have been overlooked in the decision tree community [12, 13].

Given that it is difficult to satisfy the ranking unbiasedness Property 4.1 in general, we show how α in our adjustment $Ar^2(\alpha)$ might be used to tune the bias when it is possible. Figure 6 shows that with big α ($\alpha \approx 0.4$) relationships on small n have higher probability to be selected. On the other hand, small α ($\alpha \approx 0.05$) tunes the bias towards higher sample size n . On a real ranking task, it is reasonable to rank according to $Ar^2(\alpha)$ and see how the rank changes with changes of α rather than relying on a single ranking based on biased measures such as r^2 , Ar^2 , or Sr^2 . The best value for α can be chosen by cross-validation when it is possible. Similar conclusions can be drawn for MIC and its adjusted versions.

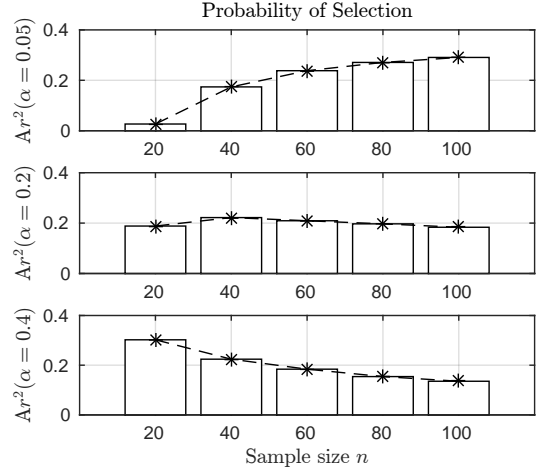


Figure 6: Probability of selection of a sample \mathcal{S}_n when X is linearly related to Y with 10% white noise using $Ar^2(\alpha)$: α tunes the bias towards small n with a big α (bottom plot) or big n with a small α (top plot).

4.2 Experiments with Pearson Correlation and MIC. MIC and r^2 have been used in [1] to identify the strongest related pair of socio-economic variables using the WHO dataset. This dataset is a collection of $m = 357$ variables for $n = 201$ countries. Some of the variables have a high percentage of missing values and they are available on much fewer than $n = 201$ samples. In this section, we aim to alert the users of MIC and r^2 about ranking biases for relationships induced on different sample size n . We conduct an experiment: we choose a reference socio-economic variable Y and select

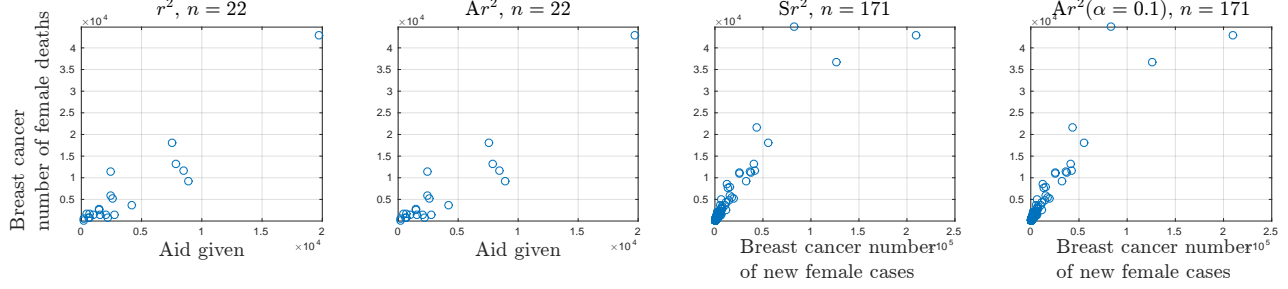


Figure 7: Plot of the top-most dependent variable X to $Y = \text{“Breast cancer number of female deaths”}$ according different adjustments for r^2 . r^2 and Ar^2 favor relationships on small n . Sr^2 , and $Ar^2(\alpha = 0.1)$ penalize relationships on small n and select more reasonably $X = \text{“Breast cancer number of new female cases”}$.

the top related variable according to r^2 and its adjusted versions. Then, we estimate the dependency between two variables based on the data points available for both X and Y . We only consider dependencies estimated on at least $n \geq 10$ data points. Figure 7 shows the top-most dependent variable X to $Y = \text{“Breast cancer number of female deaths”}$ using r^2 , Ar^2 , Sr^2 , and $Ar^2(\alpha = 0.1)$. The top-most dependent variable according to r^2 and Ar^2 is $X = \text{“Aid given”}$ which quantifies the amount of aid given to poor countries in million US\$. Instead, Sr^2 and $Ar^2(\alpha = 0.1)$ identify $X = \text{“Breast cancer number of new female cases”}$ which seems a more reasonable choice given that the number of deaths might be correlated with new cancer cases. Indeed as seen in the previous Section, r^2 and Ar^2 favour variables induced on small n . Moreover from the plot in Figure 7 we see that they are very sensitive to extreme values or outliers: i.e. the United States show a very high number of deaths due to breast cancer $\approx 43,000$ in a year and a very high amount of aid given ≈ 20 Billion US\$; this increases the chances for a high r^2 or Ar^2 .

MIC is even more inclined to select variables induced on small n . For example we see in Figure 8 that if we target $Y = \text{“Maternal mortality”}$ which quantifies the number of female deaths during pregnancy (out of 100,000 live births), and we choose MIC or AMIC to identify the top dependent variable, we get $X = \text{“Oil consumption per person”}$ (tonnes per year). There seems to exist an inversely proportional relationship between X and Y , possibly due to the common cause of overall economic development but it is difficult to argue in favor of the amount of oil/energy consumption per person as the most dependent variable to maternal mortality. We also identified the top variables according to SMIC and AMIC($\alpha = 0.01$) computed with 10,000 Monte Carlo permutations. More specifically, $\text{SMIC} = \frac{\text{MIC} - \text{EMIC}_\alpha}{\text{SDMIC}_0}$, where SDMIC_0 is the unbiased estimator of the standard deviation of MIC permutations; and $\text{AMIC}(\alpha) = \text{MIC} - q_0(1 - \alpha)$, where

Table 1: Average sample size n for the top relationships in the WHO datasets. The raw estimator of a dependency measure \hat{D} favours relationships on small n . Instead, its standardized version $S\hat{D}$ favours big n . With $A\hat{D}(\alpha)$ it is possible to tune the bias towards small n (big α) or big n (small α).

Measure	r^2	MIC
\hat{D}	114.6 (min)	103.1 (min)
$A\hat{D}$	115.1	106.9
$S\hat{D}$	133.7 (max)	131.8 (max)
$A\hat{D}(\alpha = 0.4)$	116.2 (min)	111.7 (min)
$A\hat{D}(\alpha = 0.05)$	121.1	119.4
$A\hat{D}(\alpha = 0.1)$	120.2 (max)	117.4 (max)

$q_0(1 - \alpha)$ is the $\lceil (1 - \alpha) \cdot S \rceil$ -th MIC value from the sorted list of S MIC permutations in ascending order (See Appendix A.1 for more details). The top variables according to SMIC and AMIC($\alpha = 0.01$) are instead variables related to communicable/non-communicable (infectious/non-infectious) diseases which is more intuitively related to mortality.

Table 1 shows the average sample size n for the chosen top variables with different adjustments. The user of dependency measures should be aware of the bias of raw dependency estimators \hat{D} towards small n and try to explore results from their adjusted versions $S\hat{D}$ and $A\hat{D}(\alpha)$ when ranking. Ultimately, the latter can be chosen to tune the bias towards smaller n (big α) or big n (small α).

4.3 Experiments with Gini gain in Random Forests. Splitting criteria are known to be biased towards variables induced on small n or categorical with many categories. Standardized measures and p -values are the state-of-the-art strategy to solve this problem [7, 12, 13, 9]. However, we saw that standardized measures are unbiased in ranking only when the population value $\mathcal{D}(X, Y) = 0$, and the user might better

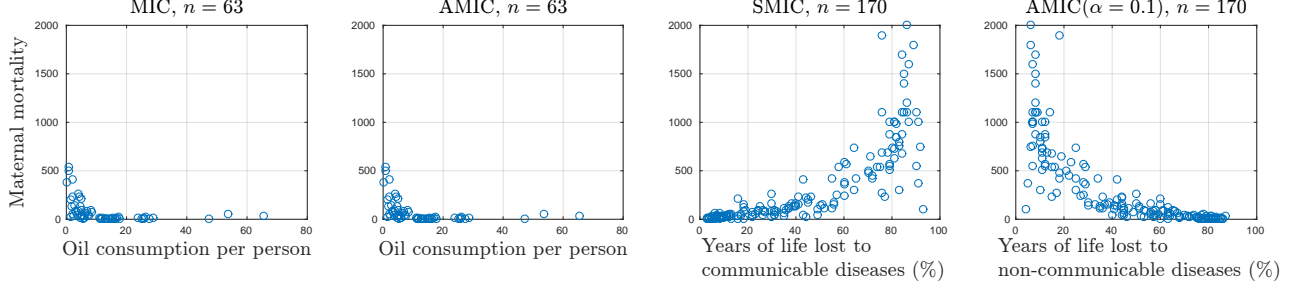


Figure 8: Plot of the top-most dependent variable X to $Y = \text{“Maternal Mortality”}$ according different adjustments for MIC. MIC and AMIC are biased towards small n . SMIC and AMIC($\alpha = 0.1$) select more reasonably either $X = \text{“Years of life lost to communicable diseases”}$ or $X = \text{“Years of life lost to non-communicable diseases”}$.

tune the bias using the parameter α . The optimal α can be found with cross-validation.

Here we use the expected value $E_0[\text{Gini}]$ and the variance $\text{Var}_0(\text{Gini})$ of Gini proposed in [7] to standardize Gini gain as per Definition 4.1 (See Appendix A.2 for more details). Moreover, we employ them to compute the adjusted Gini gain $\text{AGini}(\alpha)$ as follows:

PROPOSITION 4.1. *The adjustment for ranking at level $\alpha \in (0, 1]$ for Gini gain is:*

$$\text{AGini}(\mathcal{S}_n|X, Y)(\alpha) = \text{Gini}(\mathcal{S}_n|X, Y) - \tilde{q}_0(1 - \alpha)$$

where $\tilde{q}_0(1 - \alpha)$ is an upper bound for the $(1 - \alpha)$ -quantile of Gini gain equal to:

$$E[\text{Gini}_0(\mathcal{S}_n|X, Y)] + \sqrt{\frac{1 - \alpha}{\alpha} \text{Var}(\text{Gini}_0(\mathcal{S}_n|X, Y))}.$$

The proof of this upper bound is proposed in the supplement A.2.

We compare WEKA random forests with Gini, SGini, and $\text{AGini}(\alpha)$ as splitting criteria. To our knowledge this is the first time SGini and $\text{AGini}(\alpha)$ are tested in random forest. The forest is built on 1,000 trees taking care of sampling data with no replacement (50% training set records for each tree) to not introduce further biases towards categorical variables with many categories [14]. We employed 17 UCI datasets and 2 datasets with many categorical variables studied in [15]. The latter datasets are related to biological classification problems and some of the variables can take as many categories as the number of amino acids at a given site in a viral protein: e.g. in the HIV dataset, there exist variables which can take 21 possible values and induce splits of 21-cardinality in the trees. Table 2 shows the AUC performance of random forest computed with 50 bootstrap 2-fold cross-validation using different splitting criteria. All our adjustments improve on the AUC of the random forest built with Gini. We fixed α in $\text{AGini}(\alpha)$ to show that using a value of 0.05

or 0.1 on average increases the random forest’s AUC: see Figure 9. Moreover, we also tuned α with cross-

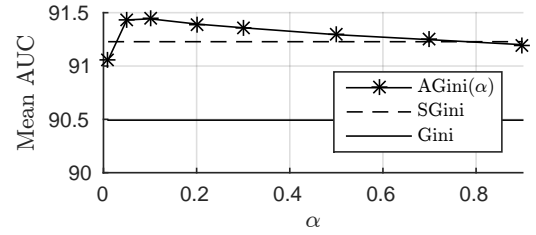


Figure 9: AUC of random forest varying α : with $\alpha = \{0.01, 0.05\}$ it achieves the best results on average.

validation for the best performance, where small and big α correspond to penalization of variables with big and small number of categories respectively. Indeed, the performance of random forests with $\text{AGini}(\alpha)$ with α tuned is statistically better than the one built with Gini according to the 1-sided Wilcoxon signed rank test: $p\text{-value} = 0.0086$. Although the observed effect size is small, it was consistent, and there is no extra computational effort. We strongly believe that adjusted splitting criteria are beneficial given that i) they can be plugged in random forests where Gini is currently used to improve classification accuracy on data sets with categorical variables or with missing values, ii) they exhibit the same computational complexity as the Gini, and iii) they are easy to implement, in particular much easier than the estimation of their confidence interval with a possibilistic loss function proposed recently [16].

5 Conclusion

In this paper we discussed how to adjust dependency measure estimates between two variables X and Y using the null hypothesis of their independence. This is particularly important to achieve *interpretable quantification* of the amount of dependency. For this task, we proposed the quantification adjusted measures Ar^2

Table 2: Random forest AUC using different splitting criteria. Either (+), (=), or (−) means statistically greater, equal, or smaller according to the 1-sided paired *t*-test at level 0.05 than random forest AUC with Gini gain.

Dataset	Variable with max number of categories	Number of classes	$m_{\text{categorical}} +$ $m_{\text{continuous}} =$ m	n	Gini	SGini	AGini ($\alpha = 0.05$)	AGini(α) with α tuned
Credit-g	11	2	13 + 7 = 20	1000	77.47	78.17 (+)	77.66 (=)	78.16 (+)
australian	14	2	8 + 6 = 14	690	92.59	93.09 (+)	93.02 (+)	93.11 (+)
bio-promoters	4	2	57 + 0 = 57	106	97.03	97.29 (+)	97.41 (+)	97.53 (+)
flags	14	8	26 + 2 = 28	194	90.49	91.75 (+)	91.75 (+)	91.83 (+)
kr-vs-kp	3	2	36 + 0 = 36	3196	99.86	99.86 (=)	99.86 (=)	99.86 (=)
led7	2	10	7 + 0 = 7	3200	94.18	94.18 (=)	94.18 (=)	94.18 (=)
lymph	8	4	15 + 3 = 18	148	92.91	93.16 (+)	93.13 (=)	93.13 (=)
mfeat-pixel	7	10	240 + 0 = 240	2000	99.58	99.63 (+)	99.64 (+)	99.64 (+)
mito	21	2	23 + 0 = 23	175	79.32	79.28 (=)	79.26 (=)	79.10 (=)
monks1	4	2	6 + 0 = 6	556	99.96	99.85 (−)	97.38 (−)	99.78 (−)
monks2	4	2	6 + 0 = 6	601	64.86	70.89 (+)	77.83 (+)	80.72 (+)
monks3	4	2	6 + 0 = 6	554	98.73	98.74 (=)	98.74 (=)	98.73 (=)
solar-flare	6	6	11 + 0 = 11	323	89.17	89.23 (+)	89.22 (=)	89.23 (+)
splice	6	3	60 + 0 = 60	3190	99.52	99.52 (=)	99.52 (=)	99.52 (=)
steel	2	2	6 + 27 = 33	1941	99.94	99.93 (−)	99.93 (−)	99.94 (−)
tae	2	3	2 + 3 = 5	151	72.25	72.33 (+)	73.23 (+)	73.65 (+)
tic-tac-toe	3	2	9 + 0 = 9	958	97.83	97.93 (+)	97.95 (+)	97.94 (+)
c-to-u	5	2	42 + 3 = 45	2694	89.74	89.42 (−)	89.28 (−)	89.61 (−)
HIV	21	2	1030 + 0 = 1030	355	84.08	89.27 (+)	89.58 (+)	89.58 (+)
<i>p</i> -value for the 1-tailed Wilcoxon signed rank test against random forest with Gini						0.0114	0.0295	0.0086

and AMIC. However, quantification adjustment is not enough to achieve *accurate ranking* of dependencies. In particular, it is very difficult to achieve ranking unbiasedness. In this task, the user should explore the possible rankings obtained with standardized and ranking adjusted measures, varying the parameter α . We demonstrated that our Sr^2 , $Ar^2(\alpha)$, SMIC, and $AGini(\alpha)$ can be used to obtain more meaningful rankings, and that $AGini(\alpha)$ yields higher accuracy in random forests. The code for our measures, experiments, and supplementary material have been made available online¹.

Acknowledgments: Supported by AWS in Education Grant Award and ARC FT110100112.

References

- [1] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti, “Detecting novel associations in large data sets,” *Science*, vol. 334, no. 6062, pp. 1518–1524, 2011.
- [2] I. Kononenko, “On biases in estimating multi-valued attributes,” in *IJCAI*, 1995, pp. 1034–1040.
- [3] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.
- [4] S. Romano, N. X. Vinh, J. Bailey, and K. Verspoor, “Adjusting for chance clustering comparison measures,” *arXiv preprint arXiv:1512.01286*, 2015.
- [5] L. Breiman, “Random forests,” in *Machine Learning*, vol. 45, no. 1, 2001, pp. 5–32.
- [6] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed., 2011.
- [7] A. Dobra and J. Gehrke, “Bias correction in classification tree construction,” in *ICML*, 2001, pp. 90–97.
- [8] M. Reimherr, D. L. Nicolae *et al.*, “On quantifying dependence: A framework for developing interpretable measures,” *Statistical Science*, vol. 28, no. 1, pp. 116–130, 2013.
- [9] S. Romano, J. Bailey, V. Nguyen, and K. Verspoor, “Standardized mutual information for clustering comparisons: one step further in adjustment for chance,” in *ICML*, 2014, pp. 1143–1151.
- [10] D. Giles. More on the distribution of r-squared. [Online]. Available: <http://davegiles.blogspot.com.au/2013/10/more-on-distribution-of-r-squared.html>
- [11] P. I. Good, *Permutation, parametric and bootstrap tests of hypotheses*. Springer, 2005, vol. 3.
- [12] C. Strobl, A.-L. Boulesteix, and T. Augustin, “Unbiased split selection for classification trees based on the gini index,” *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 483–501, 2007.
- [13] E. Frank and I. H. Witten, “Using a permutation test for attribute selection in decision trees,” in *ICML*, 1998, pp. 152–160.
- [14] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, “Bias in random forest variable importance measures: Illustrations, sources and a solution,” *BMC Bioinformatics*, vol. 8, 2007.
- [15] A. Altmann, L. Tolosi, O. Sander, and T. Lengauer, “Permutation importance: a corrected feature importance measure,” *Bioinformatics*, vol. 26, no. 10, pp. 1340–1347, 2010.
- [16] M. Serrurier and H. Prade, “Entropy evaluation based on confidence intervals of frequency estimates: Application to the learning of decision trees,” in *ICML*, 2015, pp. 1576–1584.

¹<https://sites.google.com/site/adjdep/>

Supplementary Material

A Dependency Measure Estimators

Here we formally define the dependency measure estimator of Gini gain between two categorical variables X and Y and the Maximal Information Coefficient (MIC) [1] on a sample \mathcal{S}_n .

Gini gain and mutual information (a.k.a. information gain)

Let X be a categorical variable with r possible values, then n_i^X with $i = 1, \dots, r$ is the count of records with value i for X in the sample \mathcal{S}_n . Similarly, let Y be a categorical variable with c possible values, n_j^Y with $j = 1, \dots, c$ is the count of records with value j for Y . Finally, the count of records for the pairs that associate the values i for X and j for Y is denoted as n_{ij} . Gini gain are estimated on the empirical probabilities $\frac{n_{ij}}{n}$, $\frac{n_i^X}{n}$, and $\frac{n_j^Y}{n}$ which can be stored in a contingency table:

		Y				
		n_1^Y	\dots	n_j^Y	\dots	n_c^Y
X	n_1^X	n_{11}	\dots	\cdot	\dots	n_{1c}
	\vdots	\vdots		\vdots		\vdots
	n_i^X	\cdot		n_{ij}		\cdot
	\vdots	\vdots		\vdots		\vdots
	n_r^X	n_{r1}	\dots	\cdot	\dots	n_{rc}

Figure 10: $r \times c$ contingency table that stores the bivariate frequency distribution of X and Y for the sample \mathcal{S}_n .

Gini gain is defined as:

(A.1)

$$\text{Gini}(\mathcal{S}_n|X, Y) \triangleq 1 - \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2 - \sum_{i=1}^r \frac{n_i^X}{n} \left(1 - \sum_{j=1}^c \left(\frac{n_{ij}}{n_i^X}\right)^2\right)$$

Instead, the mutual information (MI) between X and Y is defined as:

$$(A.2) \quad \text{MI}(\mathcal{S}_n|X, Y) \triangleq \sum_{i=1}^r \sum_{j=1}^c \frac{n_{ij}}{n} \log_2 \frac{n_{ij} \cdot n}{n_i^X n_j^Y}$$

Maximal Information Coefficient (MIC)

Given a sample \mathcal{S}_n from the continuous variables X and Y , MIC is the maximal normalized MI computed across all the possible $r \times c$ grids to estimate the bivariate frequency distribution of X and Y . Each $r \times c$ discretizes the scatter plot of X and Y in $r \cdot c$ bins to compute their

frequency distribution:

(A.3)

$$\text{MIC}(\mathcal{S}_n|X, Y) \triangleq \max_{r \times c \text{ grids with } r \cdot c \leq n^a} \frac{\text{MI}(\mathcal{S}_n|X, Y)}{\log_2 \min\{r, c\}}$$

where a is a parameter often set to 0.6 [1].

A.1 Distribution of MIC under the null. The distribution of $\text{MIC}_0(\mathcal{S}_n|X, Y)$ can be computed using $s = 1, \dots, S$ Monte Carlo instances $\text{MIC}_0^{(s)}$ of MIC computed on the sample $\mathcal{S}_n^0 = \{(x_{\sigma_x(k)}, y_{\sigma_y(k)})\}$ obtained by permutations of the sample \mathcal{S}_n : $\sigma_x(k)$ and $\sigma_y(k)$ are the permuted indexes of the points x_k and y_k respectively. The expected value of MIC under the null can be estimated with:

$$(A.4) \quad \text{EMIC}_0 = \frac{1}{S} \sum_{s=1}^S \text{MIC}_0^{(s)}$$

The standard deviation of MIC under the null can be estimated with:

$$(A.5) \quad \text{SDMIC}_0 = \sqrt{\frac{1}{S-1} \sum_{s=1}^S (\text{MIC}_0^{(s)} - \text{EMIC}_0)^2}$$

A.2 Distribution of Gini gain under the null.

The analytical distribution of Gini gain in Eq. (A.1) is difficult to compute. Nonetheless, it is possible to compute its expected value and variance. According [7] the expected value of Gini gain under the null using the multinomial model is:

$$(A.6) \quad E[\text{Gini}_0(\mathcal{S}_n|X, Y)] = \frac{r-1}{n} \left(1 - \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2\right)$$

and the variance $\text{Var}[\text{Gini}_0(\mathcal{S}_n|X, Y)]$ is:

$$(A.7) \quad \frac{1}{n^2} \left[(r-1) \left(2 \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2 + 2 \left(\sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2 \right)^2 - 4 \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^3 \right) \right. \\ \left. + \left(\sum_{i=1}^r \frac{1}{n_i^X} - 2 \frac{r}{n} + \frac{1}{n} \right) \times \right. \\ \left. \left(- 2 \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2 - 6 \left(\sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^2 \right)^2 + 8 \sum_{j=1}^c \left(\frac{n_j^Y}{n}\right)^3 \right) \right]$$

We can compute the $(1 - \alpha)$ -quantile of Gini gain under the null using its expected value and variance:

PROPOSITION 4.1. *The adjustment for ranking at level $\alpha \in (0, 1]$ for Gini gain is:*

$$\text{AGini}(\mathcal{S}_n|X, Y)(\alpha) = \text{Gini}(\mathcal{S}_n|X, Y) - \tilde{q}_0(1 - \alpha)$$

where $\tilde{q}_0(1 - \alpha)$ is an upper bound for the $(1 - \alpha)$ -quantile of Gini gain equal to:

$$E[\text{Gini}_0(\mathcal{S}_n|X, Y)] + \sqrt{\frac{1 - \alpha}{\alpha} \text{Var}(\text{Gini}_0(\mathcal{S}_n|X, Y))}.$$

Proof. Let μ and σ be the expected value and standard deviation respectively. We apply the Cantelli's inequality to find an upper bound for $q_0(1 - \alpha)$: $P(\text{Gini} \leq \mu + \lambda\sigma) \geq \frac{\lambda^2}{1 + \lambda^2}$ for $\lambda \geq 0$. If we set $\frac{\lambda^2}{1 + \lambda^2} = \alpha$ then $P(\text{Gini} \leq \mu + \sqrt{\frac{1 - \alpha}{\alpha}}\sigma) \geq \alpha$. This implies $q_0(1 - \alpha) \leq \mu + \sqrt{\frac{1 - \alpha}{\alpha}}\sigma$. \square